Title: Quantum Computing

Author(s): Pakin, Scott D.

Intended for: Web

Issued: 2017-12-20

**Disclaimer:**
Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

# Quantum Computing

By Scott Pakin

A *bit* can be

**true** or **false**

**left** or **right**

**up** or **down**

**1** or **0**

**set** or **reset**

**on** or **off**

**high** or **low**

It is the most primitive unit of information

**yes** or **no**

**in** or **out**

# $N$ bits can represent any one of $2^N$ values

➡ 000

001

010

011

➡ 100

101

110

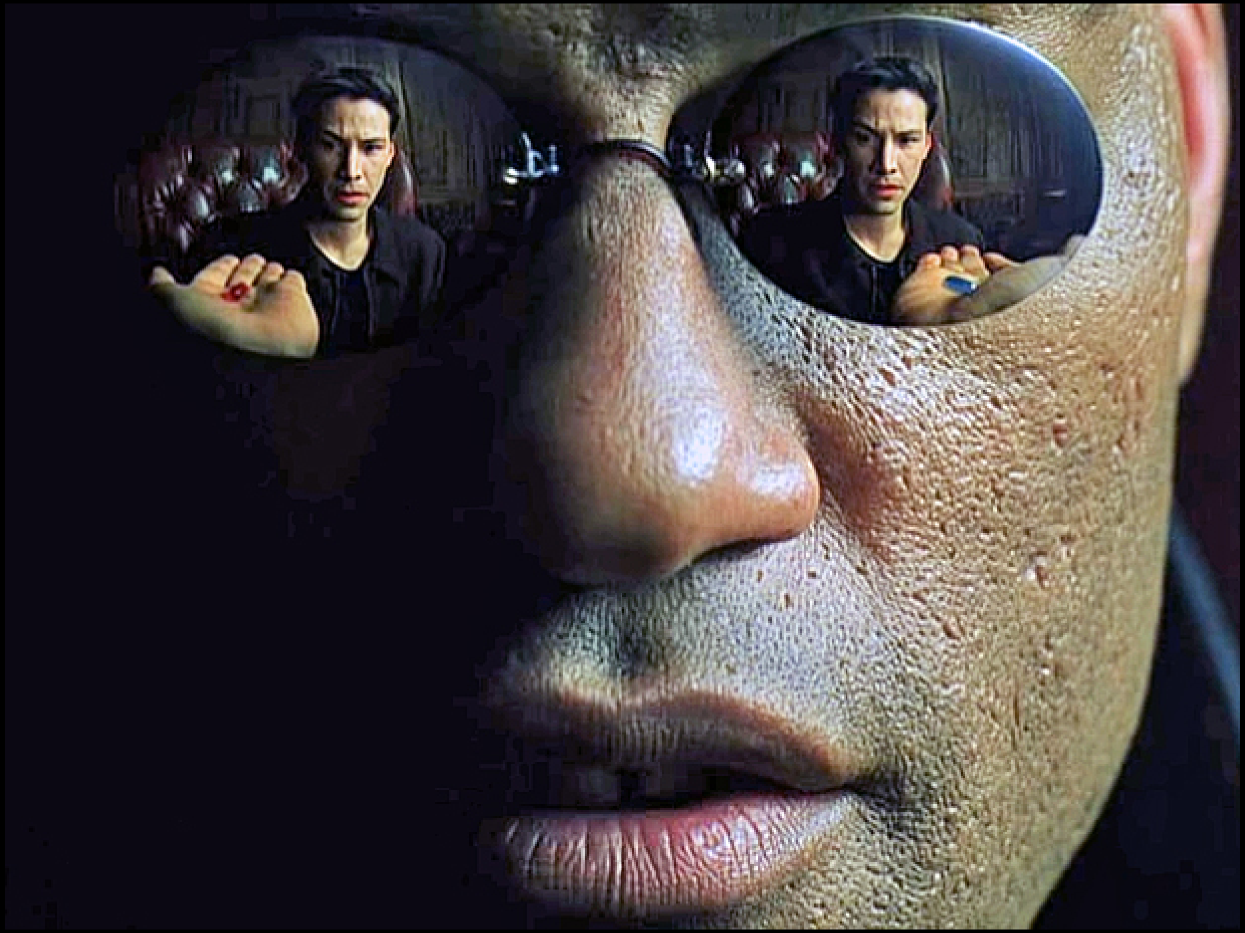
111

There are **four** possible 1-bit operators

- $a$
- $\bot$
- $\top$
- $\neg a$

# There are **sixteen** possible 2-bit operators

- ⊥
- a↓b
- a↚b
- ¬a
- a↛b
- ¬b
- a⊻b
- a↑b
- a∧b
- a↔b
- b
- a→b
- a
- a←b
- a∨b
- T



You take the blue pill, the story ends; you wake up in your bed and believe whatever you want to believe. You take the red pill, you stay in Wonderland, and I show you how deep the rabbit hole goes.

# A *qubit* is a point in a 2-D Hilbert space

(i.e., a pair of complex numbers)

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} \qquad \begin{pmatrix} 0 \\ -1 \end{pmatrix} \qquad \begin{pmatrix} i - \frac{3}{2} \\ -\frac{\sqrt{12i - 1}}{2} \end{pmatrix}$$

# A qubit's state as a linear combination of basis vectors:

How much "0-ness"

How much "1-ness"

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

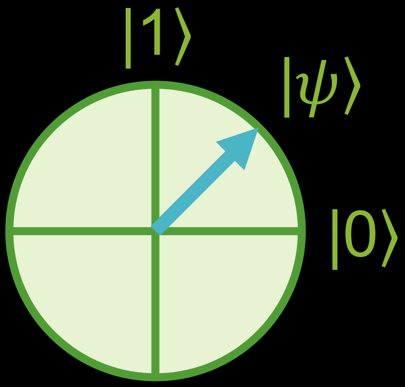
$$|0\rangle \qquad |1\rangle$$

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$\alpha, \beta \in \mathbb{C}$$

$$|\alpha|^2 + |\beta|^2 = 1$$

# A qubit can simultaneously have properties of both 0 and 1

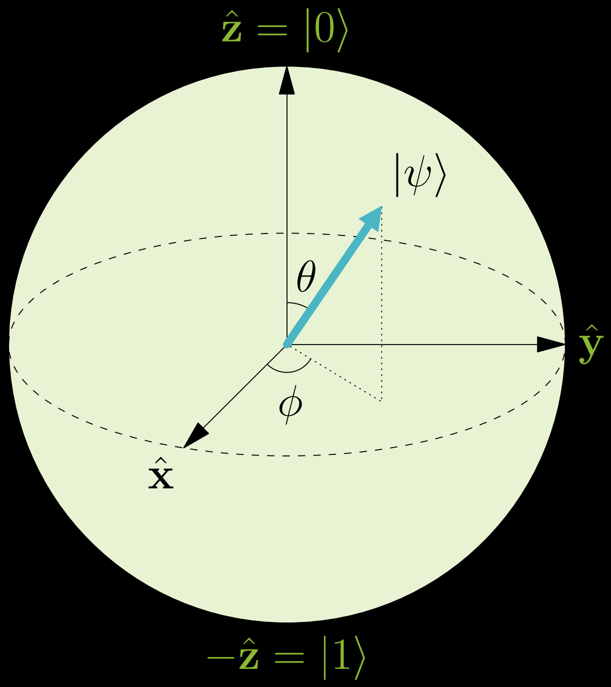


We call $|\psi\rangle$ a *superposition* of 0 and 1

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

# There are an infinite number of 0s and 1s

Easier to discern from the *Bloch sphere*, a commonly used projective vector space



All $e^{i\phi}|0\rangle$ represent different *phases* of 0

$N$ qubits can represent **all** $2^N$ values simultaneously

- ➡ 000
- ➡ 001
- ➡ 010
- ➡ 011
- ➡ 100
- ➡ 101
- ➡ 110
- ➡ 111

$N$ qubits are represented with a vector of length $2^N$

# Measuring a qubit collapses it to a classical 0 or 1

$\alpha|0\rangle + \beta|1\rangle$ is measured as 0 with probability $|\alpha|^2$ and 1 with probability $|\beta|^2$

**Oracle**: I'd ask you to sit down, but, you're not going to anyway. And don't worry about the vase.
**Neo**: What vase?

[*Crash*]

**Oracle**: That vase.

…

**Neo**: How did you know?
**Oracle**: Ohh, what's really going to bake your noodle later on is, would you still have broken it if I hadn't said anything?

# A 2-qubit state can be constructed from the tensor product of two 1-qubit states

(and similarly for $N$-qubit states)

$$|10\rangle = |1\rangle \otimes |0\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \cdot 1 \\ 0 \cdot 0 \\ 1 \cdot 1 \\ 1 \cdot 0 \end{pmatrix} = \begin{array}{c} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \end{array} \begin{array}{l} \text{00-ness} \\ \text{01-ness} \\ \text{10-ness} \\ \text{11-ness} \end{array}$$

# Qubits do not necessarily have their own identity

$\begin{pmatrix}\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2}\end{pmatrix}^{\mathbf{T}}$ is read as 00, 01, 10, or 11 with 25% probability apiece

$\begin{pmatrix}\mathbf{0} & \frac{\mathbf{1}}{\sqrt{\mathbf{2}}} & \frac{\mathbf{1}}{\sqrt{\mathbf{2}}} & \mathbf{0}\end{pmatrix}^{\mathbf{T}}$ is read as 01 or 10 with 50% probability apiece

Measuring/modifying one qubit affects the other

We call this *entanglement*

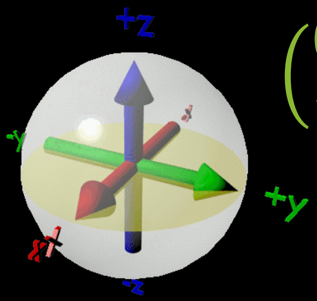
# There are **infinitely many** 1-qubit operators

(2×2 unitary matrices)

reversible

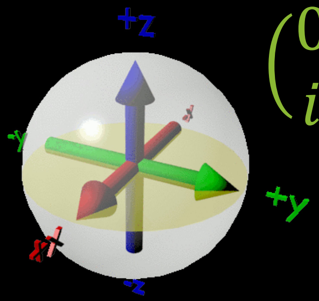
- Example #1: $NOT = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
- Example #2: $\sqrt{NOT} = \frac{1}{2}\begin{pmatrix} 1+i & 1-i \\ 1-i & 1+i \end{pmatrix}$

# Visualizing some 1-qubit operators (“gates”)



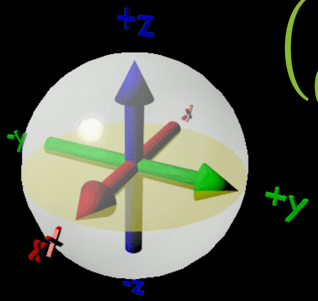
$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Pauli *x* gate (*X*)



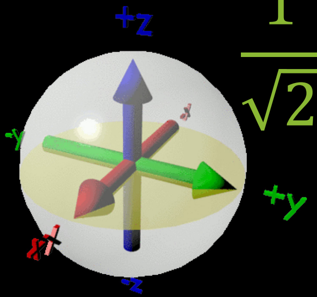
$$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

Pauli *y* gate (*Y*)



$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Pauli *z* gate (*Z*)



$$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Hadamard gate (*H*)

- A Hadamard gate puts a qubit in a perfect superposition of 0 and 1
- $XX = YY = ZZ = HH = I$
- Implication: deterministic → random → deterministic

# There are **infinitely many** 2-qubit operators

(4×4 unitary matrices)

- Example #2:

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

- Example #3:

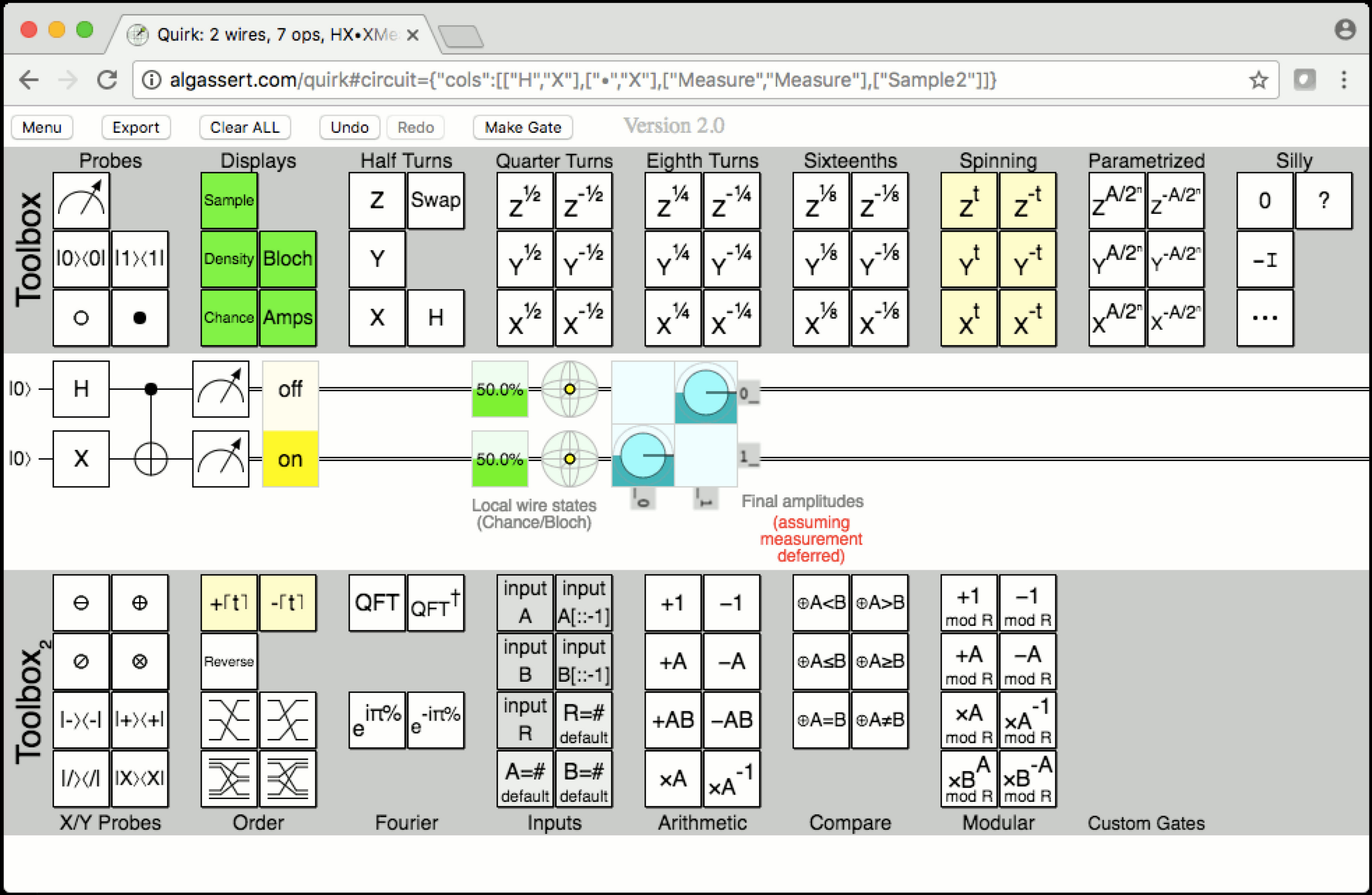
$$SWAP = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$iSWAP = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & i & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Negate b if a = |1⟩
- Swaps the values of qubits a and b
- Useful for entangling
- Can construct [illegible] gates
- What if b is in a superposition?
- What if a is in a superposition?

I know what you’re thinking, ’cause right now I’m thinking the same thing.  Actually, I’ve been thinking it ever since I got here: Why, oh why, didn’t I take the **blue** pill?

# Quantum circuits

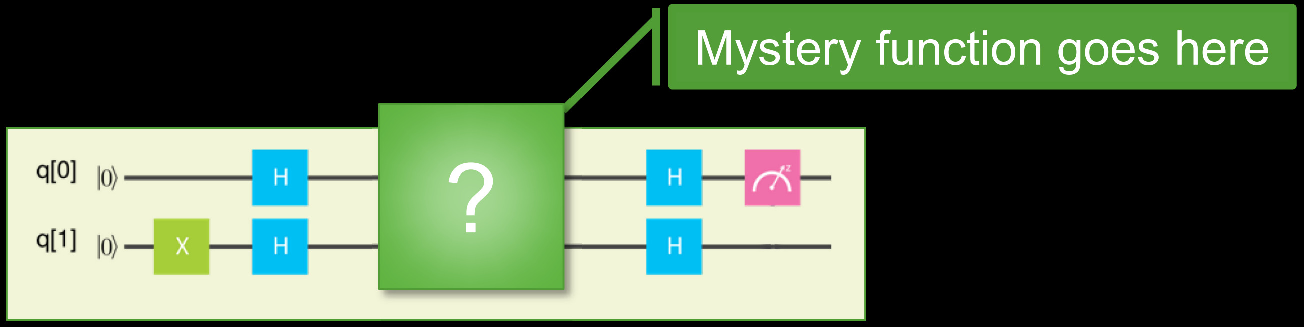


Quirk: 2 wires, 7 ops, HX•XMe

algassert.com/quirk#circuit={"cols":[["H","X"],["•","X"],["Measure","Measure"],["Sample2"]]}

# What’s the big deal?

- First answered by Deutsch and Josza in 1992



Mystery function goes here

- Determine if a given black-box function is constant or balanced
  - For one bit, constant functions are $f(x)=0$ and $f(x)=1$; balanced are $f(x)=x$ and $f(x)=\neg x$
  - *Classical*: Evaluate $f(x)$ twice
  - *Quantum*: Evaluate $f(x)$ once—returns 0 for balanced, 1 for constant
- Increasing performance improvement with scale
  - *Classical*: Evaluate $f(x)$ $\lfloor N/2+1 \rfloor$ times for $N$ bits
  - *Quantum*: Evaluate $f(x)$ once for $N$ bits

# Quantum algorithms

- Begin and end classically (i.e., only $|0\rangle$ and $|1\rangle$ states)
- Quantum in between
- Can compute on all $2^N$ combinations in parallel
- The catch: Only one $N$-bit answer comes out

# Challenges

- Reduce/cancel out probability amplitudes of non-solutions
- Manage rotations in an $N$-dimensional Hilbert space
- To date, only a small number of algorithms exist

| **Speedup over classical** | **#** |
|---|---|
| Exponential | 2 |
| Superpolynomial | 27 |
| Polynomial | 25 |
| Constant | 2 |
| Varies | 4 |
| *Total* | *60* |

Stephen Jordan
*Quantum Algorithm Zoo*
http://math.nist.gov/quantum/zoo

# Unordered search

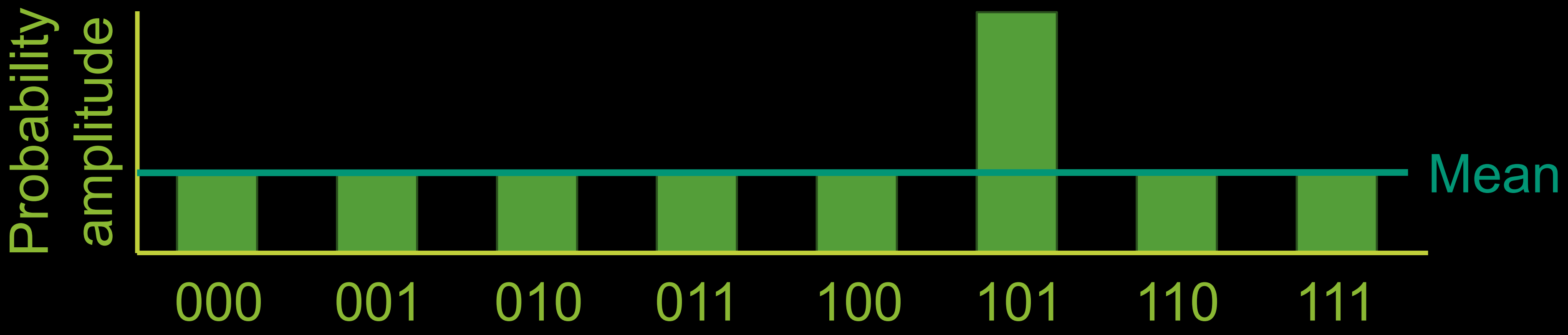
- Which screen’s image matches a given pattern?
- Classical: $O(N)$ queries
- Quantum: $O(\sqrt{N})$ queries (next two slides)

# Grover's search algorithm

- Given
  - A power-of-2 number of elements
  - A guarantee that exactly one element matches the pattern
  - An operator $U_\omega$ that, given an element $|x\rangle$, flips the probability amplitude iff the element matches (i.e., $U_\omega|x\rangle = -|x\rangle$ for $x = \omega$ and $U_\omega|x\rangle = |x\rangle$ for $x \neq \omega$)
- Return the matching element

# Grover's search algorithm

- **Approach:** For $\sqrt{N}$ iterations, alternately apply $U_\omega$ followed by "Grover diffusion operator" $U_s$
- $U_s \equiv 2|s\rangle\langle s| - I$, which flips amplitudes around the mean

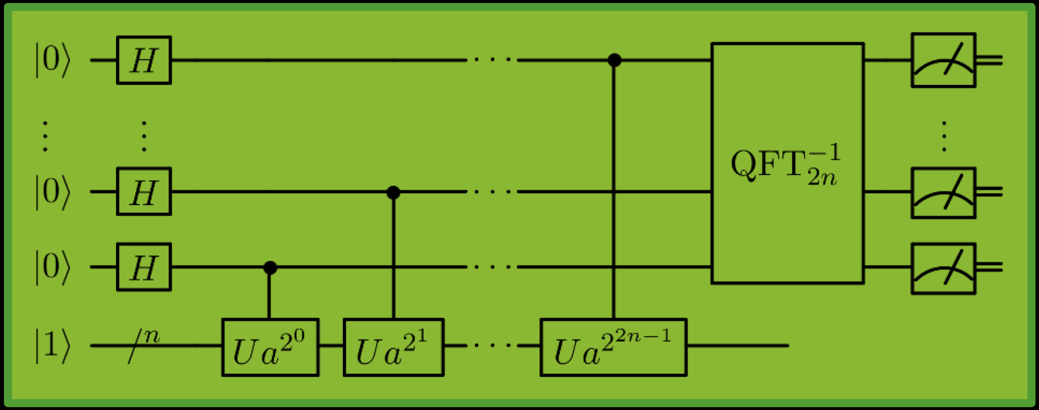


# Integer factorization

- Factor an integer into a product of two primes
- Best known classical algorithm has running time $O(2^{\sqrt[3]{N}})$
- Best known quantum algorithm has running time $O(\log^3 N)$ (next slide)
- Exponential speedup
- Expected that factoring a 50-100 bit number would be intractable classically but tractable with quantum: “quantum advantage”

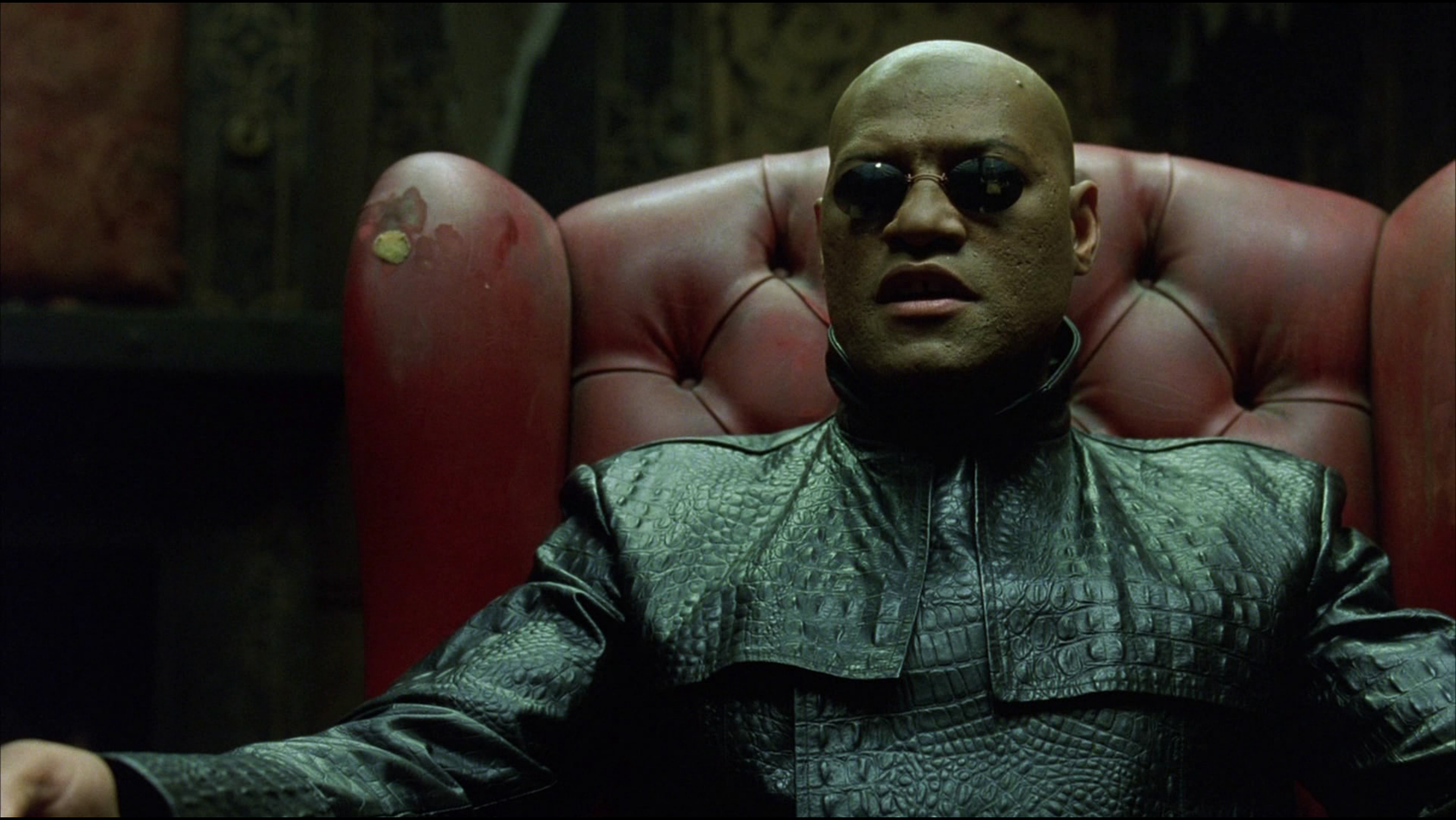
# Shor's algorithm

- It's not too hard to factor $N$ into primes $p$ and $q$ if we know the period of the sequence {$a^1$ mod $N$, $a^2$ mod $N$, $a^3$ mod $N$, …} for some $a<N$ with $p$ł$a$ and $q$ł$a$
- Apply an inverse quantum Fourier transform to find the period
- All else is classical—and randomized



# Conclusions

- *Very* different form of computing
- Qubits carry more information than classical bits (e.g., phase)
- Quantum gates perform state transformations in high-D spaces
- Exploit superpositioning and entanglement for full parallelism
- Manipulate probability amplitudes to isolate correct answers
- Potential for *exponential* performance improvement



I’m trying to free your mind, Neo.  But I can only show you the door.  You’re the one that has to walk through it.